Artificial Intelligence and Law **7:** 289–301, 1999.

Analogy making in legal reasoning with neural networks and fuzzy logic

JÜRGEN HOLLATZ
Siemens AG, Corporate Technology, Munich, Germany

Abstract. Analogy making from examples is a central task in intelligent system behavior. A lot of real world problems involve analogy making and generalization. Research investigates these questions by building computer models of human thinking concepts. These concepts can be divided into high level approaches as used in cognitive science and low level models as used in neural networks. Applications range over the spectrum of recognition, categorization and analogy reasoning. A major part of legal reasoning could be formally interpreted as an analogy making process. Because it is not the same as reasoning in mathematics or the physical sciences, it is necessary to use a method, which incorporates first the ability to specify likelihood and second the opportunity of including known court decisions. We use for modelling the analogy making process in legal reasoning neural networks and fuzzy systems. In the first part of the paper a neural network is described to identify precedents of immaterial damages. The second application presents a fuzzy system for determining the required waiting period after traffic accidents. Both examples demonstrate how to model reasoning in legal applications analogous to recent decisions: first, by learning a system with court decisions, and second, by analyzing, modelling and testing the decision making with a fuzzy system.

1. Legal Reasoning with Neural Networks

In legal reasoning, the judge follows rules defined by the written law, but also includes precedents in his decision process. Often the interpretation of the law varies to a large extent among judges and it is difficult to find a common ground.

In this paper, a problem was chosen where it is not possible to directly apply the written law at all and judges have to rely to a large extent on precedents. This legal methodology of analyzing prototype decisions is very similar to the neural network philosophy: The neural network learns to make a prediction on a new case given these prototypical cases.

In general, there are two difficulties: (1) representing the law in rule-like sentences (“if … then … ”) which could be entered in a computer, and (2) describing the circumstances in parametric form. Both require the expert’s experience. For a more general treatment of this question see Philipps (1989). In the problem described here, we had the advantage that data (precedents) were available in tabulated, parametric form, and we had a set of rules defined by a legal expert.

The algorithm described provides the ability to translate legal decision making into a computational form containing the following steps: Select the most essential features for the problem as inputs for the network. Enter the prototypical cases into the network and learn by adopting the parameters. Afterwards the network will be able to produce a reasonable output for a new constellation that is not prototypical. Due to its receptive field properties the network effectively compares a certain input to the most analogous cases.

The neural network provides also the possibility of pruning the inputs and rules. The networks architecture is variable. In principle, after training the rules can be extracted again from the network, but we have to make sure that the set of rules is as concise as possible, otherwise the value of the extracted rules is limited. We would like to find the smallest number of rules that can still describe the knowledge sufficiently. Also the network should be encouraged to find the smallest number of conjuncts, which in this case means that a basis function (rule) is only dependent on a minimal amount of inputs. Both constraints can be implemented through additional penalty terms to the quadratic cost function. For this technique see Hollatz (1992). For the legal example, this pruning procedure produced a quite optimal set of rules and conjuncts.

In this particular application we examine the identification of precedents in the area of the law of immaterial damages (see also Philipps et al. 1989). As input, several circumstances (type, seriousness and duration of injury, seriousness and duration of consequences, sex, impairment of occupation, particular severity, medical malpractice) are used. The amount of compensation of immaterial damages in German Marks is given as output. Our data base consisted of 200 court decisions. These cases including the factual circumstances and the legal consequences are available in tabulated format from the German automobile club (ADAC 1991). Our system was required to learn to predict the magnitude of expected compensation of immaterial damages. Because of the relatively large and consistent data base, this problem is well-suited to being solved with a neural approach.

1.1. THE NEURAL NETWORK

In the following the neural network and its mathematical description is given. For a more detailed description of neural networks see Hertz et al. (1991), for example. We consider networks that describe a mapping from an input space $\mathbf{x} \in \Re^n$ (the facts of the case in legal reasoning) to an output space $y \in \Re$ (the final decision). In the widest sense, we consider a fuzzy rule to be domain-specific knowledge about the same input/output mapping, which can be quantified in simple expressions of the form: *if* (premise) *then* (conclusion), where the premise makes a statement about the input space and the conclusion makes a statement about the output space. In this section, we first describe the network architecture and then show how to initialize the network with fuzzy rules.

Let $\{Rule_i, i{:}1 \ldots M\}$ be a set of fuzzy rules. For every rule we introduce a basis function $b_i(\mathbf{x})$ which is equal to one wherever the premise of the rule is valid, and equal to zero otherwise. Alternatively, we define a number (typically but not necessarily between 0 and 1) which we call the validity of the rule and which indicates the certainty with which a particular rule can be applied, given the input **x**. We assume that a basis function can be described by a multivariate Gaussian

$$b_i(\mathbf{x}) = \kappa_i \, exp\left[-\frac{1}{2}(\mathbf{x}-\mathbf{c}_i)^t \Sigma_i^{-1}(\mathbf{x}-\mathbf{c}_i)\right], \tag{1}$$

where we assume that the covariance matrix Σ_i is diagonal and where σ_{ij} is the jth diagonal element in Σ_i. In the following, we further assume that $\kappa_i = 1$. The parameter c_{ij} defines the position in the jth dimension of the input space where $Rule_i$ has the largest validity. The parameter $range_{ij} = \sigma_{ij}$ indicates approximately the range in which $Rule_i$ is valid in the jth input dimension. We say that each hidden unit has its own *receptive field* in the input space, a region centered on c_{ij}. The width of this receptive field is proportional to σ_j. A basis function approximately corresponds to a fuzzy logic membership function. We assign a parameter or weight w_i to every basis function which is equal to the (expected) value of y, given that $Rule_i$ is valid. We define the network output to be

$$y_N(\mathbf{x}) = \frac{\sum_i w_i b_i(\mathbf{x})}{\sum_i b_i(\mathbf{x})}. \tag{2}$$

In regions where only one rule demonstrates significant validity, the output is equal to w_i. In regions where more than one rule has a significant validity, the equation forms a weighted average of the outputs of those rules. Note that the resulting architecture described by Equation (2) is identical to the neural network architectures used by Moody and Darken (1989). When training data arrive, the network architecture can be modified, and the network parameters (centers, widths, and weights) can be adapted using backpropagation. The topology of the network is shown in Figure 1.

Given the state of the input the network makes a prediction. If we can measure the actual state of output we can adjust the network such that the prediction improves. If we use the quadratic cost function

$$E_D = 1/2 \sum_k (y^k - y_N(\mathbf{x}^k))^2 = 1/2 \sum (e^k)^2, \tag{3}$$

with pattern number k, network output $y_N(\mathbf{x}^k)$ and the desired output y^k, the network can be adjusted by gradient descent, where

$$\partial E_D/\partial q_i = -\sum_k e^k \frac{b_i(\mathbf{x}^k)}{\sum_j b_j(\mathbf{x}^k)},$$

if q_i is parameter w_i and

$$\partial E_D/\partial q_i = -\sum_k e^k \frac{w_i - y_N(\mathbf{x}^k)}{\sum_j b_j(\mathbf{x}^k)} \frac{\partial}{\partial q_i} b_i(\mathbf{x}^k),$$

RULE 11:				
IF	number of injured body parts	=	0.45	(~ 3 injured body parts)
	highest severity of injury	=	0.13	(~ only small injuries like abrasions)
	highest duration of injury	=	0.14	(~ short (< 2 weeks))
	diminishment of earning capacity	=	0.00	(~ no diminishment of earning capacity)
	duration of diminishment	=	-0.33	(~ short (< 2 weeks))
	sex	=	0.46	(~ not mentioned in decision)
	impairment of occupation	=	0.00	(~ no impairment of occupation)
	particular severity	=	0.16	(~ no particular severity)
	medical malpractice	=	0.00	(~ no medical malpractice)
THEN	amount of immaterial damages	=	5.31 DM	

RULE 14:				
IF	number of injured body parts	=	0.40	(~ 3 injured body parts)
	highest severity of injury	=	1.00	(~ amputation of at least one body part)
	highest duration of injury	=	1.01	(~ permanent)
	diminishment of earning capacity	=	0.99	(~ 99%)
	duration of diminishment	=	0.92	(~ permanent)
	sex	=	0.47	(~ not mentioned in decision)
	impairment of occupation	=	0.00	(~ no impairment of occupation)
	particular severity	=	0.36	(~ no particular severity)
	medical malpractice	=	0.23	(~ no medical malpractice)
THEN	amount of immaterial damages	=	1007.86 DM	

Figure 1. Two typical rules extracted from the network after learning.

if q_i is a parameter in b_i (center c_{ij}, range σ_{ij}). These gradients could be inserted into the formula to adjust the parameters with a certain learning rate η. The parameter change is given by:

$$\Delta q_i = -\eta \cdot \frac{\partial E_D}{\partial q_i}. \tag{4}$$

The functional equality between radial basis function networks and fuzzy inference systems is given, if the number of the radial basis function nodes (receptive fields) is equal the number of if-then-rules. The output of every fuzzy if-then rule is either a constant or a function dependent on the input vector. The membership functions of every single rule were chosen as gaussian functions and the t-norm operator is the multiplication. The radial basis function network as well as the fuzzy inference system use the same method to compute the output: the weighted sum. As explained above, fuzzy rules can be expressed as neural networks. The transformation method is demonstrated in the following example. If the linguistic variable is represented as the center of the corresponding membership function, a rule could be expressed as follows:

IF $[((x_1 = A)$ AND $(x_4 = B)]$ OR $(x_2 = C)$

THEN $y = d \times x^2$.

In this example A, B and C are linguistic variables, which represent fuzzy sets with gaussian membership functions and the centers a, b and c. The fuzzy rule is coded in the network as:

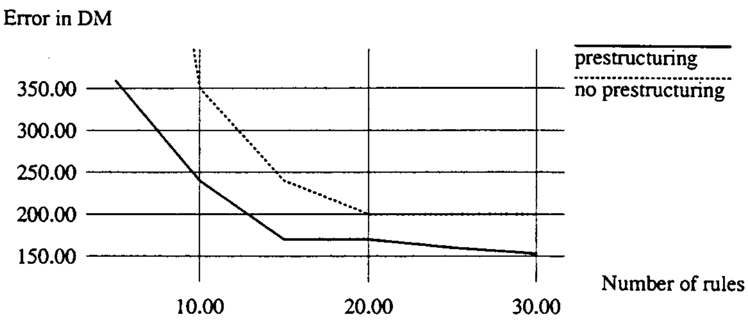

Figure 2. Mean error of the network with and without prior knowledge in the generalization phase.

$$\text{premise}_i: \quad b_i(\mathbf{x}) = \exp\left[-\frac{1}{2}\frac{(x_1-a)^2+(x_4-b)^2}{\sigma^2}\right] + \exp\left[-\frac{1}{2}\frac{(x_2-c)^2}{\sigma^2}\right]$$

$$\text{conclusion}_i: \quad w_i(\mathbf{x}) = d \times x^2.$$

This formulation is related to the fuzzy logic approach of Takagi et al. (1985). The connection between fuzzy membership functions and Gaussian basis functions is examined by Wang and Mendel (1992).

1.2. RESULTS

We tried two different experiments. First, the network learns without prestructuring, and after the learning phase rules were extracted and analyzed. In the second experiment, the improvement in generalization ability due to the prestructuring of the network was measured as a function of the number of rules (hidden units) which were used to prestructure the network. During network training, rule refinement takes place and after the training phase, it is possible to extract rules as explanation components for decision processes.

In the first experiment, we want to find rules extracted from the given data set, in this case the 200 court decisions. As described in Section 2, we used a two-layer feed-forward network with gaussian radial basis units and with normalized network output (Equation (2)). The architecture used consisted of 9 input, 15 hidden, and 1 output unit. The parameters A^i_j and B^i were updated by a gradient descent learning rule. After 1000 cycles, the rules were extracted and presented as shown in Figure 1.

The premises are connected with AND-operators and its values A^i_j are normalized to the interval [0; 1]. All ranges r_{ij} are fixed to 0.25 and not updated. The

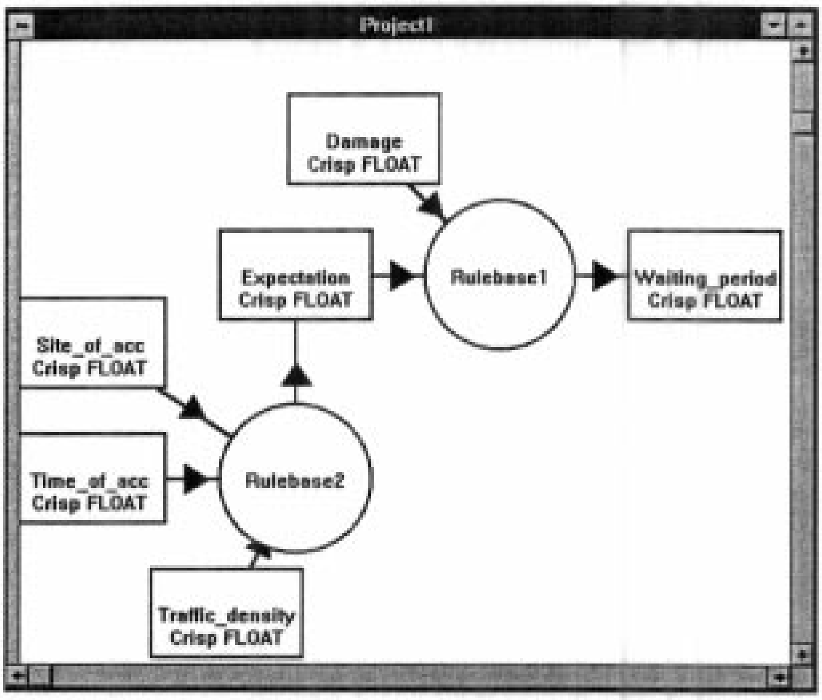

Figure 3. Architecture of the fuzzy expert system.

conclusion B^i is the real unnormalized value given in German Marks. Most rules could be interpreted as easily as the ones shown in Figure 2: $Rule_{11}$ is an example of a rule which covers cases with low compensation and $Rule_{14}$ is an example for a rule with high compensation. After rule extraction we could conclude, for example, the following: (1) not the number, but the severity of the injury is important for the decision finding process, and also (2) that reduction of earning capacity is an important factor. But we could also conclude (3) that the decision is independent of sex (0 = female, 0.5 not mentioned in decision, 1 = male). For some rules the interpretation is more difficult, which can be partially attributed to the fact that court decisions are often somewhat contradictory.

Figure 3 compares the generalization ability of networks with and without prestructuring as a function of the number of hidden units (rules). The same network structure as in the previous experiment was used. 180 patterns were given in the training set and 20 patterns were used for testing the generalization ability. Prestructuring the network consisted of presetting the values of A^i_j and B^i. It can be seen that prestructured networks learn faster and generalize better due to the additional knowledge used in network training.

2. Legal Reasoning Using Fuzzy Systems

According to classical logic an object either is or is not a member of a class. In fuzzy logic the either-or classification does not exist because fuzzy logic is based on the idea that an element can also be part of a class to a certain degree. The membership degree takes values in the range of [0, 1]. This approach offers the ability to process imprecise concepts and uncertain information of human reasoning. As vague legal concepts are often used in German law, fuzzy logic has a really great future in jurisprudence, which will be shown by the example of 'determining the required waiting period after traffic accidents'.

The purpose of a fuzzy logic application in legal science is to assist lawyers and judges in forming a judgement on facts given by the computer. Examples of assistance systems already exist in road traffic law in the form of quota tables which contain the level of responsibility and the distribution of damage between the parties involved in a traffic accident (Krumbholz et al. 1988). Additionally, there is a fuzzy expert system to define the legal animal owner (§833 BGB (civil code)) who is liable for bodily injury, death or property damage caused by the animal (Heithecker 1993).

According to German law (§142 I StGB (criminal code) *leaving the scene of an accident*) a person involved in a traffic accident must not leave the place of an accident. We should distinguish between two completely different situations of accident: In the first case, there are other people at the scene of an accident who are interested in clearing up the facts of the accident. In the second case, nobody else is present at the scene of an accident because the car driver caused an accident without involving other road users. For instance, the car driver damaged another parking car while he was leaving the parked place (Poeck 1994).

In this case the behaviour of the person involved in a traffic accident will only be correct if the person waits for an *'amount of time adequate under the cirumstances'* until someone arrives who is willing to take down his name and details concerning the car and the accident. The person is allowed to leave the scene of the accident when the required waiting period has expired and nobody has arrived (Dreher and Tröndle 1995).

Neither the statute nor commentaries nor court decisions nor publications tell us what amount of time is adequate under what circumstances (Haft 1991; Philipps 1993).

The fuzzy logic expert system is able to treat the above mentioned second case.

3. The Fuzzy System

Vague legal concepts like 'short', 'medium' or 'long' waiting periods can be represented by fuzzy sets, because the terms merge with one another without clear boundaries. Fuzzy logic offers the ability to analyse the linguistic usage of legal terms and to incorporate them into a fuzzy expert system (Philipps 1993).

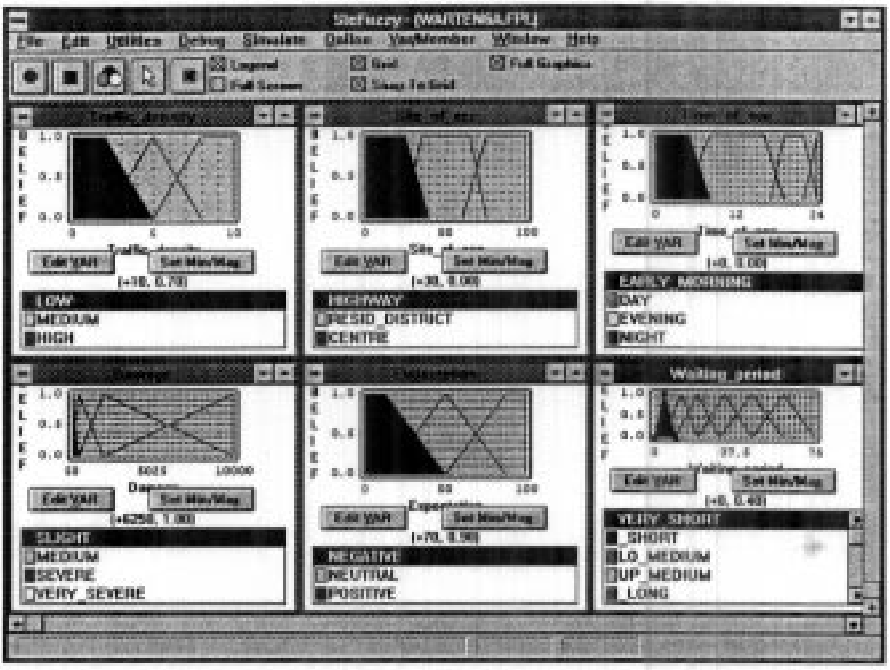

Figure 4. Variables and membership functions.

The development of the fuzzy system for determining the required waiting period is part of a master thesis (Schretter 1995) and was previously discussed in Schretter and Hollatz (1995). A prototype was created with the fuzzy tool SieFuzzy – a windows-based tool for analysis and design of fuzzy systems (SieFuzzy 1993).

The duration of the required wait is determined by the following factors (Gerathewohl 1987; Philipps 1993; Poeck 1994)

– Amount of damage
– Expectation level of someone's arrival
– Site of accident
– Time of accident
– Traffic density

According to juridical thinking the architecture of the fuzzy expert system shown in Figure 4 is structured in two stages.

The components of the expert system are described in Figure 4 and will be explained in the following:

3.1. AMOUNT OF DAMAGE

The amount of damage only includes property damage, because discussion about personal injuries would be too complex. The amount of damage is divided into the following fuzzy sets (Poeck 1994):

- Damage less than under 50 DM (German marks) is not important (Dreher and Tröndle 1995).
- Damage is *slight* between 50 and 500 DM.
- The area of *medium* damage contains values between 50 and 2000 DM.
- Damage is *severe* over 500 DM and *very severe* over 2000 DM.

3.2. EXPECTATION LEVEL OF SOMEONE'S ARRIVAL

The level of expectation means that there is a prognosis, whether someone will arrive at the scene who is willing to clear up the facts and circumstances of the accident and to inform the party whose property was damaged, or other people involved in the accident (Poeck 1994). The passer-by, who is interested in noting the facts, can be the police, the person whose property was destroyed, as well as pedestrians, cyclists or car drivers (Dreher and Tröndle 1995).

The level of expectation can only be described linguistically by the terms *'negative'*, *'neutral'* or *'positive'*. Therefore we took a fictitious numerical scale from 1 to 100 as a basis for the level of expectation. The expectation level of someone's arrival is not something that could be directly recognized, but there are various factors determining the level of expectation. We used the site, the time of an accident and the traffic density (Gerathewohl 1987; Philipps 1993):

3.2.1. *Scene of accident*

The scene of an accident influences the level of expectation. The expectation of someone's arrival will attain a higher level if the car accident happens in a town centre rather than on a highway. Pedestrians in the town centre are usually more interested in taking down the facts of an accident than car drivers on the road who often carelessly pass the place of an accident. We propose the following categories of places (Poeck 1994):

- *Highways* including motorways, highways and roads where traffic is heavy.
- In *residential districts* there are a lot of pedestrians and car drivers.
- In town *centres* we generally find only pedestrians.

As there is only a linguistic and not a numerical description for the site of accident, we took a fictitious numerical scale from 1 to 100 as a basis.

3.2.2. *Time of accident*

At night the expectation level of a passer-by arriving is very unlikely. The time of accident is divided in the following way (Philipps 1993):

- The term *‘early in the morning’* is defined from 0.00 am to 8.00 am.
- The area of the term *‘day’* contains values between 4.00 am and 7.00 pm.
- The term *‘evening’* covers the area from 5.00 pm to 12.00 pm.
- The *night* is defined from 10.00 pm to 12.00 pm.

Time is represented by decimal industrial time.

3.2.3. *Traffic density*

If the traffic density is high on a highway, more car drivers are likely to observe the accident. The expectation level of someone’s arrival will consequently increase. We suggest the following classes of traffic density (Philipps 1993):

- *Low* density is defined from 0 to 5.
- *Medium* density contains values from 2 to 8.
- ‘*High* density’ covers the range from 5 to 10.

As the traffic density is only represented linguistically and has no numerical scale, we took a fictitious numerical scale from 1 to 10 as a basis.

3.3. THE REQUIRED WAITING PERIOD

The duration of the required wait is determined by the amount of damage and the expectation level of someone’s arrival. The intervals of waiting time are defined in the following way (Poeck 1994):

- A *very short* period from 0 to 10 minutes is not a period of waiting at all.
- A *short* waiting period is a space of time from 5 to 20 minutes.
- A *lower medium* waiting period is defined from 10 to 30 minutes.
- An *upper medium* waiting period covers a range from 20 to 45 minutes.
- A *long* waiting period is described by a space of time from 30 to 60 minutes.
- A waiting period is *very long* over 45 minutes.

3.4. RULEBASE 1

Rules for determining the duration of wait, shown in Figure 4, depend on the amount of damage and the expectation level of someone’s arrival. The rules have

Table I. Matrix of rulebase 1: damage (columns) versus expectation (rows).

Waiting period	Slight	Medium	Severe	Very severe
Negative	very short	short	lower medium	upper medium
Neutral	short	lower medium	upper medium	long
Positive	lower medium	upper medium	long	very long

Table II. Matrix of rulebase 2, if the traffic density is LOW: site of accident (columns) versus time of accident (rows).

Expectation	Highway	Resid. district	Centre
Early morning	negative	negative	negative
Day	negative	neutral	negative
Evening	negative	neutral	negative
night	negative	negative	negative

no weight, because the effect of rule weight increasing or decreasing the activation of a rule, offers us no possibility of a reasonable interpretation.

3.5. RULEBASE 2

Rules for evaluating the level of expectation are represented in the tables. Because of the reason above mentioned none of the rules carries weight. In the premises of rulebase 1 and rulebase 2, we used the minimum operator. As defuzzification strategy we chose the well-known centre of area method.

As a basis for evaluating the fuzzy system we took about 100 court decisions (Poeck 1994). Nevertheless these decisions were reduced to 22 because in most cases the judges did not give any details for the real required waiting period. The

Table III. Matrix of rulebase 2, if the traffic density is MEDIUM: site of accident (columns) versus time of accident (rows).

Expectation	Highway	Resid. district	Centre
Early morning	negative	neutral	negative
Day	neutral	positive	neutral
Evening	negative	neutral	neutral
Night	negative	neutral	negative

Table IV. Matrix of rulebase 2, if the traffic density is HIGH: scene of accident (columns) versus time of accident (rows).

Expectation	Highway	Resid. district	Centre
Early morning	neutral	neutral	neutral
Day	positive	positive	positive
Evening	neutral	positive	neutral
Night	neutral	neutral	neutral

judges simply said that the duration of wait was sufficient or was not sufficient. As an approximate basis for adjusting the system we consequently took only the facts of a sufficient waiting period.

4. Outlook

As an extension of the fuzzy system we can use further factors like the weather, conspicuousness of an accident or activities by the person involved in an accident e.g. leaving a message or informing the police immediately (Gerathewohl 1987). Depending on those circumstances the required wait can be reduced or extended. Actually, there are only a few legal applications like the fuzzy expert system for determining the required waiting period after traffic accidents. In France assistance systems at law-courts or at insurance companies, like the before mentioned quota tables for damage distribution, have already been established and have become increasingly important in Germany.

In the first part of the paper an easy way of combining fuzzy rules with the inductive learning capability is presented. The practicability of analogy reasoning is shown in two legal applications.

I would like to thank Nikola Schretter for her support in preparing the second part of the paper and implementing the fuzzy rule base.

References

Dreher, E. and Tröndle, H. 1995. *Strafgesetzbuch und Nebengesetze*, Commentary, 47th edition, Munich: C.H. Beck.

Duda, R.O. and Hart, P.E. 1973. *Pattern Classification and Scene Analysis*. New York: Wiley.

Gerathewohl, P. 1987. *Erschließung unbestimmter Rechtsbegriffe mit Hilfe des Computers. Ein Versuch am Beispiel der "angemessenen Wartezeit" bei* § *142 StGB*, doctoral thesis, University of Tübingen.

Haft, F. 1991. *Strafrecht, Besonderer Teil*, 4th edition, Munich: C.H. Beck.

Heithecker, J. 1993. *Fuzzy-Logik und der "Tierhalter"*, KI 3, pp. 7–10.

Hertz, J., Krogh, A. and Palmer, R.G. 1991. *Introduction to the Theory of Neural Computation*, Addison Wesley.

Hollatz, J. and Tresp, V. 1992. A rule-based network architecture, *Proc. of the International Conference on Artificial Neural Networks*, Brighton: ICANN 92, pp. 757–761.

Hollatz, J. 1992. Supplementing neural network learning with rule-based knowledge, *Proc. of the IEEE International Conference on Neural Networks*, Vol. III, Beijing: IJCNN-92, pp. 595–600.

Jacobs, R.A., Jordan, M.I., Nowlan, S.J., and Hinton, G.E. 1991. Adaptive mixtures of local experts, *Neural Computation* **3**, 79–87, 1991.

Krumbholz, H., Paul, H., and Brüseken, W. 1988, *Haftungsquoten bei typischen Verkehrsunfällen*, NZV, pp. 168–172.

MacKay, D. 1991. *Bayesian Modeling*. Ph.D. Thesis, Caltech.

Moody, J. and Darken, Ch. 1989. Fast learning in networks of locally-tuned processing units, *Neural Computation*, Vol. 1, Massachussetts Institute of Technology, pp. 281–294.

Philipps, L. 1989. in A. Martino (ed.), Are legal decisions based on the application of rules or prototype recognition? Legal science on the way to neural networks. In A. Martino (ed.), *Pre-Proceedings of the 3rd Intern. Conf. on Logica, Informatica*, Florence, Diritto.

Philipps, L., Brass, H. and Emmerich, Q. *A Neural Network to Identify Legal Precedents*, 9th Symposium on Legal Data Processing in Europe, (CJ-IJ Symp.).

Philipps, L. 1993. *Vague Legal Concepts an Fuzzy Logic. An Attempt to Determine the Required Period of Waiting after Traffic Accidents*, informatica e diritto, Vol. 2, pp. 37–51.

Poeck, T. 1994. *Wartepflicht und Wartedauer des* §*142 Abs.1 Nr.2 StGB. Ein Vorschlag zur Bestimmung der Wartelänge anhand rechtsprechungsanalytischer und empirischer Überlegungen*, Volk, Klaus and Lang, Peter (eds), Criminalia, Vol. 2, Frankfurt a.M., 1994, doctoral thesis, Munich: Ludwig-Maximilians-University.

Schretter, N. 1995. *Konsistenz und Vollständigkeit in Fuzzy-Regelbasen*, master thesis, Prof. Dr. W. Brauer, Munich: Institute of Computer Science, University of Technology.

Schretter, N., and Hollatz, J.: *Ein Fuzzy-Logik-Expertensystem zur Bestimmung der Wartezeit nach Verkehrsunfällen*, Fuzzy-Neuro-Systeme ’95, 3rd workshop, Darmstadt, pp. 73-80.

SieFuzzy. 1993. *User’s Manual Version 1.0*, Siemens AG.

Takagi, T. and Sugeno, M. 1985. Fuzzy identification of systems and its applications to modeling and control, *IEEE Transactions on Systems, Man and Cybernetics*, **15**(1), 116–132.

Towell, G.G., Shavlik, J.W., and Noordewier, M.O. 1990. Refinement of approximately correct domain theories by knowledge-based neural networks. In *Proceedings of the Eights National Conference on Artificial Intelligence*, MA, pp. 861–866.

Tresp, V., Hollatz, J., and Ahmad, S. 1993. Network structuring and training using rule-based knowledge, to be published in *Advances in Neural Information Processing Systems* **5**. *Über 1100 Urteile zur Höhe des Schmerzensgeldes mit den neuesten Entscheidungen deutscher und ausländischer Gerichte* (German), München: ADAC-Verlag.